Newcastle University

ePrints

Naylor J, Fellermann H, Ding Y, Mohammed W, Jakubovics N, Mukherjee J, Biggs C, Wright P, Krasnogor N. Simbiotics: A Multiscale Integrative Platform for 3D Modeling of Bacterial Populations. *ACS Synthetic Biology* 2016. DOI: 10.1021/acssynbio.6b00315.

Copyright:

ACS AuthorChoice - This is an open access article published under a Creative Commons Attribution (CC-BY) License, which permits unrestricted use, distribution and reproduction in any medium, provided the author and source are cited.

DOI link to article:

10.1021/acssynbio.6b00315

Date deposited:

30/06/2017

This work is licensed under a Creative Commons Attribution 4.0 International License

Newcastle University ePrints - eprint.ncl.ac.uk

This is an open access article published under a Creative Commons Attribution (CC-BY) License, which permits unrestricted use, distribution and reproduction in any medium, provided the author and source are cited.

ACS Synthetic Biology

Research Article

pubs.acs.org/synthbio

Simbiotics: A Multiscale Integrative Platform for 3D Modeling of Bacterial Populations

Jonathan Naylor,[†] Harold Fellermann,[†] Yuchun Ding,[†] Waleed K. Mohammed,[‡] Nicholas S. Jakubovics,[‡] Joy Mukherjee,[§] Catherine A. Biggs,[§] Phillip C. Wright,[∥] and Natalio Krasnogor*,[†]

[†]Interdisciplinary Computing and Complex Biosystems (ICOS) research group, School of Computing Science, Newcastle University, Newcastle upon Tyne NE1 7RU, U.K.

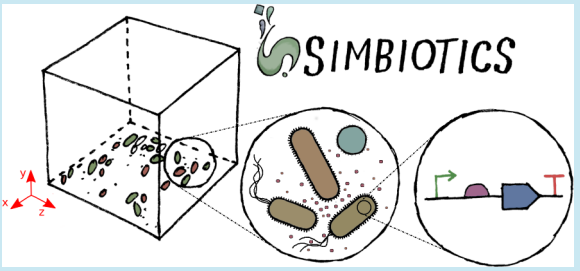
[‡]School of Dental Sciences, Newcastle University, Newcastle upon Tyne NE2 4BW, U.K.

[§]Department of Chemical and Biological Engineering, University of Sheffield, Sheffield S10 2TN, U.K.

[∥]School of Chemical Engineering and Advanced Materials, Newcastle University, Newcastle upon Tyne NE1 7RU, U.K.

S *Supporting Information*

ABSTRACT: Simbiotics is a spatially explicit multiscale modeling platform for the design, simulation and analysis of bacterial populations. Systems ranging from planktonic cells and colonies, to biofilm formation and development may be modeled. Representation of biological systems in Simbiotics is flexible, and user-defined processes may be in a variety of forms depending on desired model abstraction. Simbiotics provides a library of modules such as cell geometries, physical force dynamics, genetic circuits, metabolic pathways, chemical diffusion and cell interactions. Model defined processes are integrated and scheduled for parallel multithread and multi-CPU execution. A virtual lab provides the modeler with analysis modules and some simulated lab equipment, enabling automation of sample interaction and data collection. An extendable and modular framework allows for the platform to be updated as novel models of bacteria are developed, coupled with an intuitive user interface to allow for model definitions with minimal programming experience. Simbiotics can integrate existing standards such as SBML, and process microscopy images to initialize the 3D spatial configuration of bacteria consortia. Two case studies, used to illustrate the platform flexibility, focus on the physical properties of the biosystems modeled. These pilot case studies demonstrate Simbiotics versatility in modeling and analysis of natural systems and as a CAD tool for synthetic biology.



KEYWORDS: *bacterial population, simulation, multiscale, biofilm, agent-based model, interaction*

Bacterial colonies are networks of interacting cells that coordinate their behavior to produce system level organization.[1−3] As an additional layer of organization, bacteria may form biofilms, *i.e.*, complex communities of cells often of mixed consortia that adapt to their environment and coordinate their individual behavior.[4−6] Bacterial colonies and biofilms are relevant in many disciplines ranging from microbiology and medicine to biotechnology such as synthetic biology.[7] Numerous existing and potential future applications include antibiotic resistance, chronic infections and colonisation, dental plaque formation, biofuel production and other biological engineering applications. The pursuit of these applications relies on cell populations being understood in an integrative multiscale manner.[8,9]

Synthetic biology aims to repurpose biological components in a modular way for novel applications.[10] The development of synthetic multicellular systems allows for mixed cohorts of bacteria with each species programmed to carry out specific tasks. When developing these systems one inserts genetic devices into individual cells,[11] thus it is vital to understand how these devices effect multicellular system-wide behavior. The design, synthesis and analysis of such synthetic systems is time and resource intensive, typically involving multiple iterations around the workflow. A major challenge in the development of these systems resides in their complexity, scalability and robustness, thus models of population dynamics are essential in the realization of such systems.[12,13]

To catalyze this process, *in silico* models have been developed,[14] providing insights into the dynamics of proposed systems allowing verification of their feasibility prior to resource-intensive lab work. Such models typically focus on genetic circuits and the behavior of single cells, however the dynamics and self-organizing capabilities exhibited by populations of interacting cells is in need of additional research effort.

Understanding these multicellular systems is difficult due to the complex nature of their dynamics: first because systems emerge from the interplay between their constituent cells,

Special Issue: IWBDA 2016

Received: October 25, 2016
Published: May 5, 2017

 © XXXX American Chemical Society

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

Table 1. High-Level Feature Comparison between Simbiotics and Existing Agent-Based Modeling Tools for Bacterial Populations[a]

	Simbiotics	iDynoMiCS[24]	BSim[23]	gro[21]	CellModeller4[22]	DiSCUS[26]	Infobiotics Workbench[20]
Bacterial growth	X	X	X	X	X	X	
Rule based internal cell dynamics	X		X	X	X	X	X
Differential equation based internal cell dynamics	X		X		X	X	
SBML based internal cell dynamics	X						
Gillespie submodels	X						X
Bacterial motility	X	X	X				
Chemotaxis	X		X	X	X		
Cell surface interactions	X						
Membrane transport	X		X				X
Environmental forces	X		X			X	
Chemical environment	X	X	X	X	X		X
Extracellular polymeric substances	X	X					
2D	X	X		X	X	X	X
3D	X	X	X				
Microscopy image processing	X						
Fluid dynamics						X	
Conjugation						X	
Computational acceleration	X				X		
Programming language	Java	Java	Java	Python	Python	Python	Java

[a]X marks a feature being present.

with feedback between microprocesses such as gene regulation and metabolism, and macroprocesses such as spatial organization and differentiation.[15,16] Second, each cell has stochastic behavior that may change significantly based on its local environment,[17] understanding population behavior thus requires an in-depth knowledge of how its constituent parts behave. The population behavior of colonies can be analyzed to derive statistical correlations, however these representations do not explain the dynamics within the system or how population organization occurs.

Modeling multicellular systems in a spatially explicit manner is crucial in understanding overall behavior. The notion of spatial locality is inherent in these systems, with a cell's individual dynamics and its local environment directly effecting each other in a feedback loop. As these interactions propagate across populations we can observe how system coordination is a self-organizing property, observing higher level relationships and the driving forces behind system wide behavior.

We present Simbiotics, a multicellular bacterial simulator which represents cells as individual physical entities embedded in chemical gradients, each cell may have modeler-defined dynamics and may interact with its environment and with other cells. Through the explicit spatial modeling of individual cells we aim to both understand existing multicellular systems, especially biofilms, and offer a CAD platform to aid in the synthetic biology workflow for exploring design space and testing system robustness *in silico*, prior to synthesis.

Related Work. Several modeling tools have been developed to understand the dynamics of bacterial populations and the multicellular systems they form.[18−20] General cell population modeling tools include gro[21] and CellModeller4,[22] intended to simulate the biophysical patterning of multicellular systems in 2D, focusing on physical interactions and chemical signaling. BSim[23] is another general tool which is used to model cells in 3D, providing a general agent-based platform in which the user can define custom rules to describe cellular behavior, as well as environmental structures *via* 3D meshes. More specific tools include iDynoMiCS,[24] the successor of BacSim,[25] which is a modeling tool for biofilm formation. It allows for the specification of cellular properties and simulation of a biofilm growing on a surface. Substrate dependence is also represented in the

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

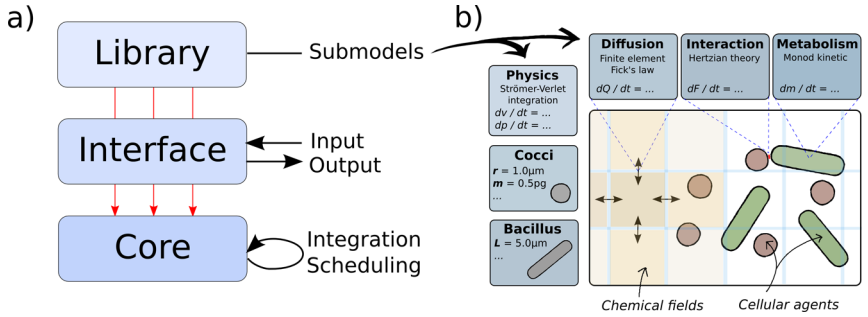


Figure 1. (a) Simbiotics architecture overview, showing that the interface is a bridge with which the user attaches library models to the simulation core. (b) A schematic of a basic Simbiotics model, composed of a set of 6 library submodels. The model specification utilitises specific library modules for each feature of the system, ranging from cell shape to metabolic behavior. The model specification can be composed and run *via* the interface layer. For a running model, the core layer integrates all model defined processes and schedules their execution for multithreaded and multi-CPU environments.

model, showing how the location of cells effects their growth. DiSCUS[26] is a specific 2D bacterial simulator modeling horizontal gene transfer between neighboring cells. All of these tools simulate chemical diffusion by discretising the environment space into subvolumes and calculating the flux between neighboring compartments.

Each platform takes a level of abstraction at which to represent the system and focuses on simulating relevant processes to the questions for which the models are developed. Existing models have a common underlying theme of an agent-based modeling framework representing cells interacting in an environment. However, a unification of these models into a common platform has not yet been conceived. Implementation of some computational acceleration such as multithreading, parallelization and GPU acceleration is also crucial for platform viability scaling to large system sizes.

These platforms may be integrated into a computational biologist or chemical engineer's workflow, used in the modeling and characterization of systems. When selecting a tool appropriate for their work, they may consider the three following factors: the modeling capacity of the platform, the usability of the platform, and the scalability of the platform. Modeling capacity is the ability of a platform to describe a given system, representing its constituent components and processes. Usability is the ease in which that description can be expressed in the platform, and scalability is the potential for the platform to simulate large industrially relevant systems.

A platform developed for this type of modeling must therefore be flexible enough to for a user to describe their target system, extendable such that new processes may be represented in the platform, usable such that the platform interface is intuitive and does not require extensive progamming to achieve valid model outputs, and scalable such that modeling of populations relevant to real system is feasible. Table 1 depicts the functionality implemented in existing modeling tools and in Simbiotics.

SIMBIOTICS

We present a spatially explicit modeling platform, *Simbiotics*, which allows for the design, simulation and analysis of synthetic bacterial populations. Simbiotics simulates hybrid-models, where an agent-based model describes bacteria and other physically interacitng agents, which is coupled to a continuous chemical environment. Each agent can have many submodels describing its internal biological processes. Unique species can be defined, and large mixed populations can be simulated. Through this coupling we can observe the interplay between individual cellular processes and population level organization.

Simbiotics provides a modeling library consisting of a wide range of processes, describing physical, chemical and biological processes of bacterial population dynamics. Model specifications may be constructed by attaching library modules to the simulator, allowing for the description of populations of custom species and environmental properties. Additionally one can define analysis tools in the model specification, automating model interactions, simulated events, data collection and analysis.

The platform software architecture is flexible, users may develop models by composing library modules, only model specific processes are then simulated. Simbiotics library modules are parameterized to allow for fine-tuning for user needs. New library modules may be developed and added to the library by implementing Simbiotics interface classes, this is described in the User Manaul. The platform and library are readily extendable to add new features as necessary, ensuring the relevance of the software as computational methods and knowledge of biological processes develop.

Novel functionality in Simbiotics include the processing of microscopic images to initialize the simulation of the spatial state, additionally the integration of existing standards such as SBML ensure the accessibility and communication of models. We allow for the modeling of processes such as membrane transport through passive or active mechanisms, active cell motility due to flagellar or pili activity and chemotaxis. Simbiotics may also simulate cell surface adhesion through processes such as electrostatic and receptor-adhesin interactions. Extracellular-polymeric substances (EPS) may also be modeled, either through a particulate representation or *via* mass-spring kinetics which cause bacteria to adhere to the substratum and other cells.

The integration of these multiscale processes is a main contribution of Simbiotics, allowing for the modeling of large bacterial populations while capturing microprocesses in individual cells.

Implementation. Simbiotics is developed in Java; it utilizes the spatial representation, multithreaded and multi-CPU parallelized execution, and the 3D rendering as implemented in the Cortex3D platform.[27] The platform is designed with a modular architecture, allowing for model features to be represented as discrete components that can be readily added, removed and modified for the specific modeling application. This is achieved *via* a three component architecture

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

comprising of a simulation core, a modeling library and a modeling interface. This plug-and-play framework allows for rapid model prototyping and reiterative designs for the reification of models. The software architecture is depicted in Figure 1 (a).

The core of Simbiotics is the computational engine, integrating all model defined processes and scheduling commands for parallel execution. It is a framework with interfaces which are populated from the modeling library.

The modeling library contains a collection of modules, which are discrete submodels describing specific model behavior. These range from physical law integrators and chemical diffusion-reaction solvers, to bacterial geometries, cellular dynamics and boundary conditions. Modules describing virtual lab components and scheduling are also present, accompanied by analysis and data exporter modules. An exhaustive list of present library modules can be found in the User Manual in Supporting Information, here we descibed how to use and compose existing modules and how to develop news to add to the library.

The modeling interface allows the user to specify the inputs and outputs of the platform. Models can be designed by composing library modules in a JSON model specification file. Modules are parametrizable to allow for their customization. The interface also allows for the optional real-time 3D rendering of the simulation, with live graph plotters visualizing model statistics and on-the-fly analysis. Simbiotics is packaged into a stand-alone jar, which can be run from command-line. It requires a configuration file which contains the Simbiotics parameters and filepath to the JSON model file. A full description of how to compose models in Simbiotics can be found in the User Manual.

Modeling. We describe the current Simbiotics Library modules, elaborating on their functionality and the mathematics used in their calculations. These submodels are independent and can be attached to models to compose them into a full model specification. A schematic which illustrates the modeling of systems through composing library modules can be seen in Figure 1 (b).

Environment. The simulation domain describes the boundaries of the environment, it may be set to be 3D or constrained to 2D. It is continuous space with a grid systems to discretise volumes for representing chemical distributions. Domain boundary conditions may describe a solid surface and its physicochemical characteristics, periodic boundaries may also be defined such that cellular and chemical entities enter the opposing side of the domain which they leave. An escape boundary can be defined such that entities are removed from the simulation when leaving the boundary. Additionally a boundary may describe a rate with which to introduce chemicals or bacteria into the environment, modeling a chemostat or stochastic bacterial world outside of the simulation domain.

Spherical Cells. Cells may be *coccus*, represented as spheres. Each has a position vector $\mathbf{p}_i$ which represents its center as coordinates in 3D continuous space bounded within the simulation domain, a radius r_i and mass m_i. Additionally each cell has a velocity vector $\mathbf{v}_i$ which describes its current velocity as a 3D vector, and a 3D unit vector which describes the orientation of the body $\hat{\boldsymbol{\psi}}_i$.

Rod-Shaped Cells. Cells may be *bacilli*, represented as rods. Each is modeled by two points at positions $\mathbf{p}_i^a$ and $\mathbf{p}_i^b$, which are connected by a rigid spring. These positions define the end points of the rod, constructing the line $\mathbf{l}_i = \mathbf{p}_i^b - \mathbf{p}_i^a$, which describes its length $l_i = |\mathbf{l}_i|$ and orientation $\hat{\boldsymbol{\psi}}_i = \frac{\mathbf{l}_i}{l_i}$. Rods are considered to be cylindrical along $\mathbf{l}_i$, with hemispherical caps. Each has a center of mass $\mathbf{p}_i$ which is the point along the rod axis that is equidistant from the two end points. Rods also have a radius r_i, mass m_i and each of its spheres has a velocity vector, $\mathbf{v}_i^a$ and $\mathbf{v}_i^b$.

A schematic for spherical and rod-shaped cells can be seen in Figure 2 (a) and (b).

Cell Neighborhood. A Verlet-list is implemented to store the nearest neighbors of a cell, for a cell i its nearest neighbor list is denoted as $\mathbf{M}_i$. A neighboring cell j is included in this list if the absolute distance between cell's closest points $\mathbf{p}_i$ and $\mathbf{p}_j$ is less than a given threshold M_r. For spherical cells, $M_r^{\mathrm{S}} = r_i + r_{\max}$, where $r_{\max}$ is the maximum cell radius in the system. For rod-shaped cells $M_r^{\mathrm{R}} = 0.5l_i + 0.5l_{\max}$, where $l_{\max}$ is the maximum rod length in the system. The total number of cells at any time t is denoted by $N(t)$. A schematic showing the representation of cellular agents can be seen in Figure 2 (f).

A cell's local environment also has chemical properties, its position $\mathbf{p}_i$ maps to a voxel V_i in the discretized grid space. This voxel contains a list of chemical species and corresponding concentrations present in that volume. The concentration at P_i may be an interpolation between V_i neighboring voxel concentrations, this is calculated with Sheppard's method as implemented in the Cx3Dp component of the software.[27] Alternatively it may be assumed that each voxel has a uniform distribution within it. A schematic can be seen in Figure 2 (c).

Physics. The motion of cells is determined by Newtonian dynamics, forces are translated into a change in velocity, and subsequently a change in velocity resulting in a change in position:

$$\frac{\mathrm{d}\mathbf{p}_i(t)}{\mathrm{d}t} = \mathbf{v}_i(t) \tag{1}$$

$$\frac{\mathrm{d}\mathbf{v}_i(t)}{\mathrm{d}t} = \frac{\mathbf{F}_i^{\mathrm{T}}(t)}{m_i} \tag{2}$$

where $\mathbf{F}_i^{\mathrm{T}}$ is the total force experienced by a bacterial cell. The equation to calculate $\mathbf{F}_i^{\mathrm{T}}$ is user defined, and may have as many force components as desired. Here we present the default equation used to calculate $\mathbf{F}_i^{\mathrm{T}}$,

$$\mathbf{F}_i^{\mathrm{T}} = \sum_{j=1}^{M_i} (\mathbf{F}_{ij}^{\mathrm{C}} + \mathbf{F}_{ij}^{\mathrm{S}} + \mathbf{F}_{ij}^{\mathrm{E}}) + \mathbf{F}_i^{\mathrm{R}} + \mathbf{F}_i^{\mathrm{F}} + \mathbf{F}_i^{\mathrm{G}} + \ldots \tag{3}$$

where i runs from 1 to $N(t)$, $\mathbf{F}_{ij}^{\mathrm{C}}$ is the force due to cell−cell collisions, $\mathbf{F}_{ij}^{\mathrm{S}}$ is the force due to specific adhesin receptor interactions, $\mathbf{F}_{ij}^{\mathrm{E}}$ is the force due to nonspecific electrostatic interactions, $\mathbf{F}_i^{\mathrm{R}}$ is the translational diffusion force, $\mathbf{F}_i^{\mathrm{F}}$ is the force of viscous drag on the cell and $\mathbf{F}_i^{\mathrm{G}}$ is the force of gravity. Together, $\mathbf{F}_i^{\mathrm{R}}$ and $\mathbf{F}_i^{\mathrm{F}}$ turn eqs 1−3 into a Langevin Dynamics approach.[28] The Strömer−Verlet method is used for the numerical integration of positions and velocities due to forces. We calculate the force components individually, as follows.

Cells experience forces due to collisions with other geometries in the 3D domain. For two cells at positions $\mathbf{p}_i$ and $\mathbf{p}_j$, $\hat{\boldsymbol{\theta}}_{ij}$ is the unit vector describing the line orientation between the cell centers from j to i, calculated as $\hat{\boldsymbol{\theta}}_{\mathrm{ij}} = \frac{\mathbf{p}_i - \mathbf{p}_j}{|\mathbf{p}_i - \mathbf{p}_j|}$.

Resolving collisions between spherical cell involves calculating response forces to apply to each cell. This is modeled as a strong spring which pushes cells apart, where $\mathbf{F}_{ij}^{\mathrm{C}}$ is the total

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

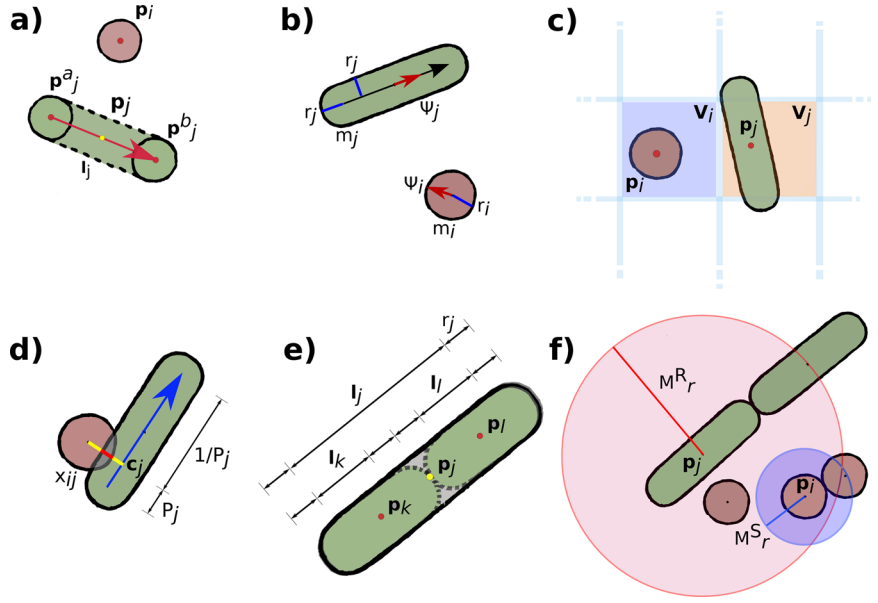


Figure 2. We consider two cells, a spherical cell (coccus) at position $\mathbf{p}_i$ and a rod-shaped cell (bacillus) at position $\mathbf{p}_j$. (a) Spheres have a position $\mathbf{p}_i$. Rods have start and end positions $\mathbf{p}_j^a$ and $\mathbf{p}_j^b$, a length $\mathbf{l}_j = \mathbf{p}_j^b - \mathbf{p}_j^a$, and a center position $\mathbf{p}_j$ which is equidistant between $\mathbf{p}_j^a$ and $\mathbf{p}_j^b$ along its length $\mathbf{l}_j$. (b) Both spheres and rods have an associated radius $r_{i/j}$ and mass $m_{i/j}$. Cocci have an orientation $\hat{\boldsymbol{\psi}}_i$, and bacilli have an orientation $\hat{\boldsymbol{\psi}}_j = \frac{\mathbf{l}_j}{l_j}$. (c) A cell's center position determines which diffusion grid voxel $V_{i/j}$ the cell is in. (d) An example of a sphere and rod cell colliding, x_{ij} is the magnitude of the overlap between cells. For a bacillus the force of a collision is distributed to its start and end points according to P_j. (e) An example of a bacillus cell dividing, where $\mathbf{p}_j$ and l_j are the position and length of the dividing cell. The positions of the child cells are $\mathbf{p}_k$ and $\mathbf{p}_l$, and they have lengths l_k and l_l. Both child cells inherent the same radius r_j. (f) An example of cell neighborhoods. The red circle represents the neighborhood range of the bacilli cell at $\mathbf{p}_j$, and the blue circle the range of the cocci cell at $\mathbf{p}_i$. For a given cell, other cells are considered in a neighbor if their center point exists within the range.

force experienced by a cell due to its colliding neighbors, K_C the spring constant for collisions and x_{ij} is the overlap distance of collision partners, $x_{ij} = r_i + r_j - |\mathbf{p}_j - \mathbf{p}_i|$,

$$\mathbf{F}_{ij}^{C} = \begin{cases} K_C x_{ij} \hat{\boldsymbol{\theta}}_{ij} & \text{if } x_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

The sphere at position $\mathbf{p}_i$ receives the force $\mathbf{F}_i^C = \mathbf{F}_{ij}^C$, and the sphere at position $\mathbf{p}_j$ receives the force $\mathbf{F}_j^C = -\mathbf{F}_{ij}^C$ according to Newton's third law.

A similar approach is taken for modeling collisions with rod-shaped cells. For the two colliding rod line segments $\mathbf{l}_i$ and $\mathbf{l}_j$, we find the closest points $\mathbf{c}_i$ and $\mathbf{c}_j$ which form the shortest line between them $\mathbf{l}_{ij} = \mathbf{c}_j - \mathbf{c}_i$. The ratio along the rods which the points lie P_i and P_j are calculated as $P_i = \frac{|\mathbf{c}_i - \mathbf{p}_i^a|}{l_i}$ and $P_j = \frac{|\mathbf{c}_j - \mathbf{p}_j^a|}{l_j}$. We calculate the overlap x_{ij} between rods to be $x_{ij} = r_i + r_j - |\mathbf{l}_{ij}|$. We then calculate the total force that rods exert on each other in the same form as eq 4, the distribution of this force onto the rod's end points follows the same approach as implemented in previous modeling work.[29] Where $\mathbf{F}_i^{Ca}$ is the force applied to point $\mathbf{p}_i^a$,

$$\mathbf{F}_i^{Ca} = -(1 - P_i)\mathbf{F}_i^C \tag{5}$$

$$\mathbf{F}_i^{Cb} = -P_i\mathbf{F}_i^C \tag{6}$$

$$\mathbf{F}_j^{Ca} = (1 - P_j)\mathbf{F}_j^C \tag{7}$$

$$\mathbf{F}_j^{Cb} = P_j\mathbf{F}_j^C \tag{8}$$

Collisions between a sphere and rod are solved as a partial form of rod−rod collisions. For a sphere at position $\mathbf{p}_i$, we find the position $\mathbf{c}_j$ on the rod line segment $\mathbf{l}_j$ which forms the shortest line between them $\mathbf{l}_{ij} = \mathbf{c}_j - \mathbf{p}_i$. We calculate the overlap x_{ij}, forces $\mathbf{F}_i$ amd $\mathbf{F}_j$ and ratio P_j in the same manner as for rod−rod collisions. The sphere receives the full force $\mathbf{F}_i$ and $\mathbf{F}_j$ is distributed onto the rods constituent spheres in the same manner as eqs 7 and 8. A schematic can be seen in Figure 2(d).

Collision force responses may be modeled with Hertzian theory rather than the force expression in eq 4. Hertzian theory models the elastic contact between colliding cells. In eq 4 $K_C x_{ij}\hat{\boldsymbol{\theta}}_{ij}$ is substituted with $E(r_i + r_j)^{1/2}x_{ij}^{3/2}$, where E is the parameter representing the elastic modulus of a cell.[30−32]

Adhesin receptor interactions are modeled as springs connecting cell geometries. An interaction between an adhesin-receptor pair q and s has a specific force constant K_{qs}^S associated with it. The extension of the spring is calculated as $\alpha_{ij} = l_a - l_r$. Where $l_a = |\mathbf{p}_i - \mathbf{p}_j|$ is the actual length of the spring, and $l_r = R_l(r_i + r_j)$ is the resting length of the spring. R_l being a spring relaxation factor allowing the spring to leave an offset between cell surfaces.

$$\mathbf{F}_{ij}^{S} = \begin{cases} K_{qs}^S \alpha_{ij} \hat{\boldsymbol{\theta}}_{ij} & \text{if } \alpha > 0 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

An adhesin-receptor interaction is reversible if a sufficiently large force pulls the cells apart. This is modeled as a maximum extension that the spring may reach before breaking. We calculate the maximum extension to be $\alpha_{ij}^{max} = C_{pq} \cdot l_r$, where C_{pq} is the extension factor for adhesin and receptor p and q. If $\alpha_{ij} > \alpha_{ij}^{max}$ the interaction spring is removed.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

Cells experience forces due to nonspecific interactions such as van der Waals interactions and electrostatic repulsion when their membranes are in close range. An established method for modeling these forces is DLVO theory.[33,34] However, this model operates on distances in the order of nanometers which are negligible in Simbiotics. We have a similar representation, modeling a proportional adhesive force as a two cell surfaces approach. K_{ij}^{E} is the adhesive force constant and d_{ij} is the distance between the cell centers defined as $d_{ij} = |\mathbf{p}_i - \mathbf{p}_j|$. Two cells interact if they are within range of each other's extended sphere of influence, defined as the cells radius r_i multiplied by a range factor r_E.

$$\mathbf{F}_{ij}^{E} = \begin{cases} \dfrac{K_{ij}^{E}}{d_{ij}^{2}}\hat{\boldsymbol{\theta}}_{ij} & \text{if } d_{ij} < \dfrac{r_E(r_i + r_j)}{2} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

To calculate the force random fluid motion has on a free-floating cell, we use eq 6 to find the force on a given particle at each moment in time. K_R is a constant describing the maximum force the cell experiences. We generate a random number between 0 and K_R and multiple it by a random unit vector $\hat{\boldsymbol{\eta}}$ to calculate the current force:

$$\mathbf{F}_i^{R} = K_R\hat{\boldsymbol{\eta}} \tag{11}$$

In the numerical integration, we must take care to normalize the force by the inverse square root of the integration step.

To describe the effect of friction for each cell we calculate a drag force which is proportional to the friction coefficient K_F, representing the viscosity of the medium. The drag force is also proportional to the velocity of the cell $\mathbf{v}_i$:

$$\mathbf{F}_i^{F} = -K_F\mathbf{v}_i \tag{12}$$

Gravity is modeled as a constant force acting on a cell,

$$\mathbf{F}_i^{G} = K_G m_i\hat{\boldsymbol{\gamma}} \tag{13}$$

where K_G is the gravitational acceleration constant, m_i is the mass of the cell and $\hat{\boldsymbol{\gamma}}$ is the unit vector describe the direction of the force, pointing to negative y.

Chemistry. Simbiotics allows for custom definition of chemical species with their respective diffusion and degradation rate constants. Chemicals can exist in the extracellular space or within cells and can be transported across membranes *via* a variety of mechanisms. Chemical reactions occur in intracellular compartments that are elaborated on in the metabolism section.

Extracellular diffusion is implemented with the finite volume method.[35] The simulation domain is decomposed into regular nonoverlapping subdomains. The flux between neighboring subdomains is calculated for each chemical species as follows:

$$J_{i\to j} = D_c\frac{S_{ij}}{d_{ij}}(u_j - u_i) \tag{14}$$

where u_i and u_j are the concentrations of a chemical species in the two neighbor subdomains, D_c is the corresponding diffusion coefficient for that chemical species and S_{ij} is the cross-section connected the two subdomains, and d_{ij} is the distance between the center points of the two subdomains.

The only extracellular reaction modeled is degradation, to calculate this a rate law can be defined for each chemical species. Where A is a chemical species and k_A is its rate of degradation:

$$A \xrightarrow{k_A} \varnothing \tag{15}$$

One may also describe chemical sources and sinks, a chemostat adjacent to any simulation domain boundary, a flux of bacteria into the domain through boundaries, and a basic flow-chamber which models a constant flow across the entire domain.

Biology. A wide range of biological processes are implemented, including cell growth kinetics and metabolic rules, cell division, motility, quorum-sensing through membrane transport, cell−cell and cell−surface adhesion as well as gene regulatory networks. Bacteria can also produce extracellular polymeric substances, which can form an extacellular matrix. We describe next the set of modeling decisions and simplifications made in order to capture and integrate these various processes.

Cell Internal Dynamics, SBML and Gillespie Integration. Intracellular processes such as gene regulation and metabolism can be modeled using either Boolean networks[36] or sets of differential equations,[37] both being widespread formalisms to specify cell internal dynamics. Neither of the two approaches makes specific assumptions of the biochemical nature of the involved components, and both can therefore be used to express arbitrary cell processes including gene regulation, metabolic reactions or high-level decision making. Additionally one may use Systems Biology Markup Language (SBML)[38] models to represent deterministic ordinary differential equations.

For Boolean network representations nodes may represent cellular states such as gene expression, phenotype differentiation or sensory information such as whether the cell is in contact with a surface. Nodes in the network are in one of the discrete states on or off, with directed arcs between nodes to describe an activation or inhibition relation. Arcs into a node are composed with propositional logic relations to form transition rules. All node transitions are solved synchronously and then updated.

Cellular processes can also be modeled with differential equations for which a solver is implemented. This solver allows for the composition of sets of ordinary differential equations that are integrated with a fourth order Runge−Kutta method. A basic Gillespie simulation module is also implemented, allowing for submodels of stochastic chemical processes. Each cell may have its own internal Gillespie model.

An SBML solver LibSBMLsim[39] is integrated, allowing for each cell to potentially have its own SBML model. An SBML model could be used to describe the metabolic or genetic dynamics of a bacterial cell. Any state variable or parameter of the SBML model can be *set* or *get* by other submodels of a cell in Simbiotics, allowing for the full integration of SBML. This enables a coupling between cell internal dynamics and interactions with their environment, such as a chemical species permeating the cell membrane and being introduced into the metabolism, or surface protein expression based on gene regulation.

Cell Growth and Death. As bacteria grow their mass increases and we calculate the change in mass Δm for the current time step based on growth and maintenance kinetics. Here we describe kinetic representations that are valid in Simbiotics, we follow a similar approach to iDynoMiCS.[4] The change in the mass of a bacterial cell is based on the calculated growth rate μ_i, where μ_i is a function of the depending nutrient concentration S_i in the local extracellular compartment V_P,

$$\frac{dm_i}{dt} = \mu_i(S_i) \tag{16}$$

Bacterial growth can be modeled as a constant process ignoring substrate dependence. Bacteria grow according to a growth rate G_r which is a uniformly distributed random variable with mean G_r and variation G_v, such that $\mu_i = G_r \pm G_v$.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

Specific growth kinetics describe how bacterial growth is calculated based on cell maintenance and available nutrient concentration. A variety of reaction kinetics are implemented which are listed in Table 2, one can compose these kinetics to design custom nutrient-based growth dynamics.

Table 2. Growth Kinetic Equations[a]

growth kinetic	equation
First-order kinetic	$\mu = G_r \pm G_v$
Monod kinetic	$\mu = \frac{S}{K_S + S}$
Simple inhibiton	$\mu = \frac{K_i}{K_i + S}$
Hill kinetic	$\mu = \frac{S^h}{K_S^h + S^h}$
Haldane kinetic	$\mu = \frac{S}{K_S + S + \frac{S^2}{K_i}}$

[a] μ is the growth rate, S is a given substance concentration, and K is the half-saturation constant of a given substance.

Cells die if their radius is below a critical threshold r_{min}. When a cell dies its geometry is completely removed from the simulation, any intracellular chemicals are then moved to the extracellular compartment in the diffusion grid which contained the cell's center of mass. Cell growth and death are implemented as an extended version of the dynamics used in the iDynoMiCS software.[24] Growth dynamics may now be coupled to either extracellular or intracellular cell chemical concentrations, additionally the distribution of intracellular chemicals upon cell death is implemented.

We assume that cell biomass density remains constant throughout the cell cycle,[40] therefore when a cell grows in mass it is expressed by a growth in volume. As a coccus cell grows, its radius r_i increases. For a bacillus cell, growth is only along the length of the cell l_i, as variations in its width are neglible in comparison.[41]

Cell Division. A binary fission library module implemented. Cell division occurs upon a cell reaching twice its original mass.[41,42] We consider child cells to inherit about half of the mass of the parent cell.[43]

Spherical Cells. This process involves dividing the geometry volume in two with a ratio D_r, which is a uniformly distributed random variable with mean 0.5 and variation D_v. This ratio describes the distribution of volume between the two child cells. The replicating bacteria will shrink and be repositioned, and a replica cell instance will be created at the calculated position of the child cell.

The volume of the two child bacteria are calculated with the division offset as follows, where V_T is the total volume of the dividing cell, V_1 and V_2 are the corresponding child cell volumes.

$$V_1 = D_r V_T \tag{17}$$

$$V_2 = (1 - D_r) V_T \tag{18}$$

All intracellular chemical molecular amounts are divided according to this ratio. Additionally all cell defined processes are copied across to the new child cell.

To find the center positions of the child bacteria, the radii r_1 and r_2 of the new cells are deduced from their volumes. A random unit vector $\hat{\mathbf{u}}$ is then generated to find an axis of division.

Rod-Shaped Cells. Rod-shaped bacteria replicate upon reaching twice their original length.[44] For a dividing rod whose center of mass is at position $\mathbf{p}_i$ and has an orientation $\hat{\boldsymbol{\psi}}_i$, we calculate the position of child cell centers of mass $\mathbf{p}_j$ and $\mathbf{p}_k$ as follows,

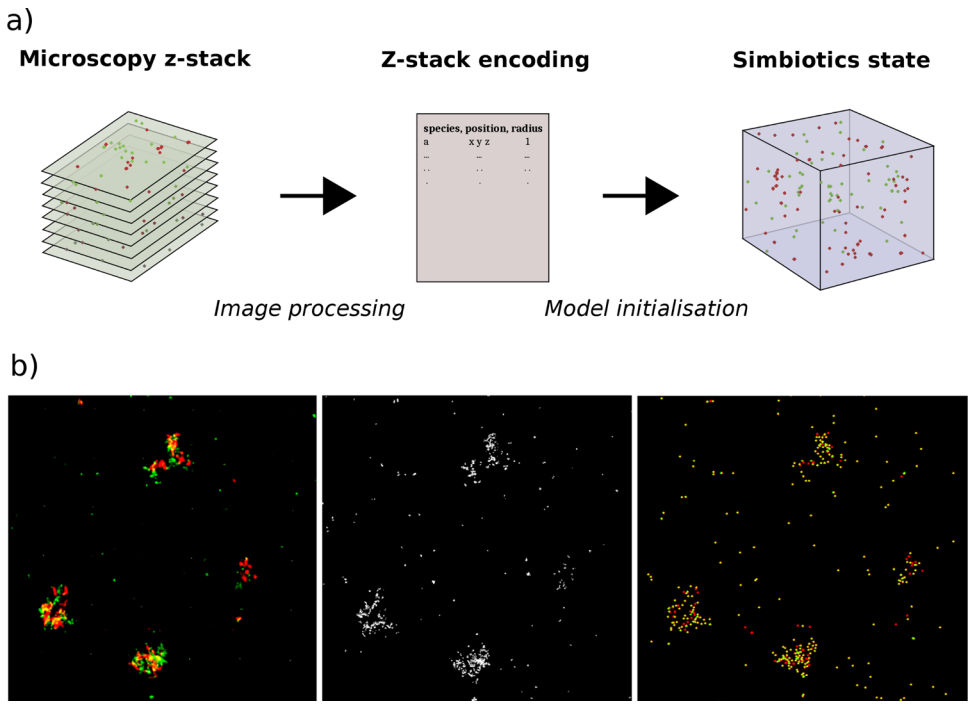


Figure 3. Microscopy image loading, showing the processing from the original microscopy Z-stack to Simbiotics model state. (a) Schematic showing that a Z-stack undergoes image processing to extract and encode features such as cell positions. This encoding is then used in Simbiotics model initialization to model state. (b) Example of microscopy image processing. Left: 2D projection of microscopy Z-stack. Middle: 2D projection of image processed Z-stack, from which image features may be extracted. Right: 2D projection of Simbiotics model, showing loaded cellular agents in the same configuration as the original Z-stack.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

$$\mathbf{p}_j = \mathbf{p}_i + (0.5l_i + r_i)\hat{\boldsymbol{\psi}}_i \tag{19}$$

$$\mathbf{p}_k = \mathbf{p}_i - (0.5l_i - r_i)\hat{\boldsymbol{\psi}}_i \tag{20}$$

Both child cells inherit the same radius as the parent cell and are of identical lengths, we must take care to subtract the radius from the child cell length, so that both child cells fit within the volume of the parent cell, $l_j = l_k = 0.5l_i - r_i$. A schematic can be seen in Figure 2 (e).

Membrane Transport. Chemicals can pass through cell membranes *via* either passive or active transport mechanisms. Passive membrane transport is solved in a similar manner as described in the Diffusion section, such that the flux is only from high to low concentrations.[45] The flux due to passive transport mechanisms for a given chemical species is denoted by J_C^P, where A_i is the surface area of the cell, C_i is the concentration of the chemical in the cell, and C_j is the concentration of the chemical in the extracellular compartment which the cell center point reside in. A membrane permeability factor for individual chemical species P_C can be defined such that the flux is proportional to a chemical's permeability:

$$J_C^P = P_C A_i (C_j - C_i) \tag{21}$$

Active transport is modeled *via* a Monod function that calculates the flux based on the source concentration, it is a unidirectional flux and the source can be set to be either intracellular or extracellular.[46] The flux due to active transport mechanisms for a given chemical species is denoted by J_C^A, where C is the chemical concentration at the source, K_C is the half-saturation constant of the chemical flux, and Q_C is the maximum flux at which the active transport mechanism can work for that chemical species:

$$J_C^A = Q_C \frac{K_C}{K_C + C} \tag{22}$$

Active Motility. Bacteria can be actively motile due to flagellar driven micromotility or pili mediated twitching-motility, these processes may be deployed to accomplish a random walk or chemotaxis.[47−49]

Micromotility in species such as *Escherichia coli* involve run and tumble phases, in which bacteria alternate between accelerating forward and rotating in place.[29] We model this by probabilities p_{endrun} and $p_{\text{endtumble}}$ with which the bacteria switch from a run or a tumble into the alternate state. During the run phase a constant force F_{endrun} is applied to the bacteria in the direction it is facing $\hat{\boldsymbol{\psi}}$. During the tumble phase we assign a new orientation $\hat{\psi}$ to the cell by generating a random unit vector. No directional force is applied to the bacteria when tumbling.

Twitching motility is modeled using the same algorithm as the micromotility with different parameters. Both p_{endrun} and $p_{\text{endtumble}}$ are relatively high, resulting in low persistence rapid movements.

Chemotaxis is modeled using a modified version of the micromotility run and tumble dynamics, implemented similar to the Keller−Segel method.[50,51] Cells perform a run and tumble and sample the concentration of the chemoattractant at periods of Δt_{memory} representing their sensory memory. Cells compare their current concentration $C(t)$ with the previous concentration they experienced $C(t - \Delta t_{\text{memory}})$. This is calculated by $C(t) - C(t - \Delta t_{\text{memory}})$, if the value is less than 1 the cell is descending the gradient and has a high probability to tumble. If the value is greater than 1 we know we are ascending a gradient or traversing a plateau, we calculate the gradient strength by how much $C(t) - C(t - \Delta t_{\text{memory}})$ is above 1. The cell has a probability to tumble p_{endrun} that is inversely proportional to the gradient strength, such that cells ascending a gradient are less likely to stop running.

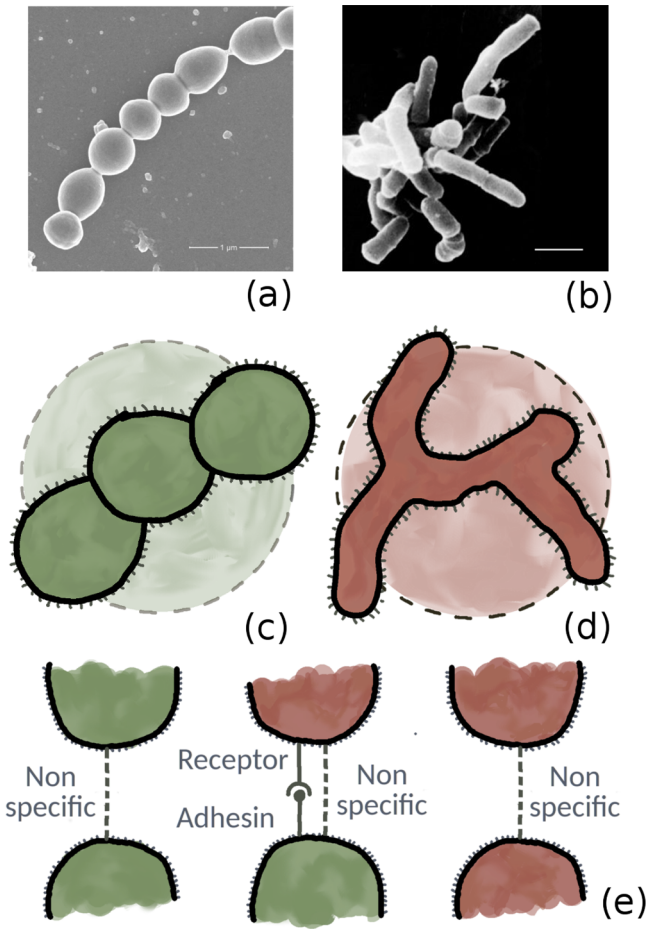


Figure 4. Microscopy images showing *Streptococcus gordonii* (a) and *Actinomyces oris* (b) where the scale bar is 1 μm. (c, d) *S. gordonii* and *A. oris* are modeled as spheres. (e) Depiction of the modeled interactions between *S. gordonii* and *A. oris*.

Extracellular Polymeric Substances. Bacteria can produce extracellular polymeric substances (EPS).[52,53] EPS can be modeled *via* two mechanisms. The first is an implicit form modeling EPS *via* mass-spring dynamics connecting adjacent cells. This implementation utilizes the same algorithm as the specific cell−surface interactions as described in the Physics section. This representation assumes that when two cells are close by their relative positions are constrained by the presence of adhesive EPS, thus a spring is formed between two neighboring geometries where the distance between their center positions $\mathbf{p}_i$ and $\mathbf{p}_j$ is less than the sum of their radii multiplied by some range factor $R_{\text{EPS}}(r_i + r_j)$.

An alternative form is to model EPS as particles that exist as geometric agents in the environment. This is modeled in a similar manner to iDynoMiCS.[24] Bacterial cells have capsular EPS which is bound to their membrane, this capsule has a volume V_i^C associated with it, and it is added to the cell's volume to calculate the cells total radius considering both active (cellular) and inactive (EPS) biomass. Upon V_i^C reaching a threshold V_{EPS}, an EPS particle is added to the local

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

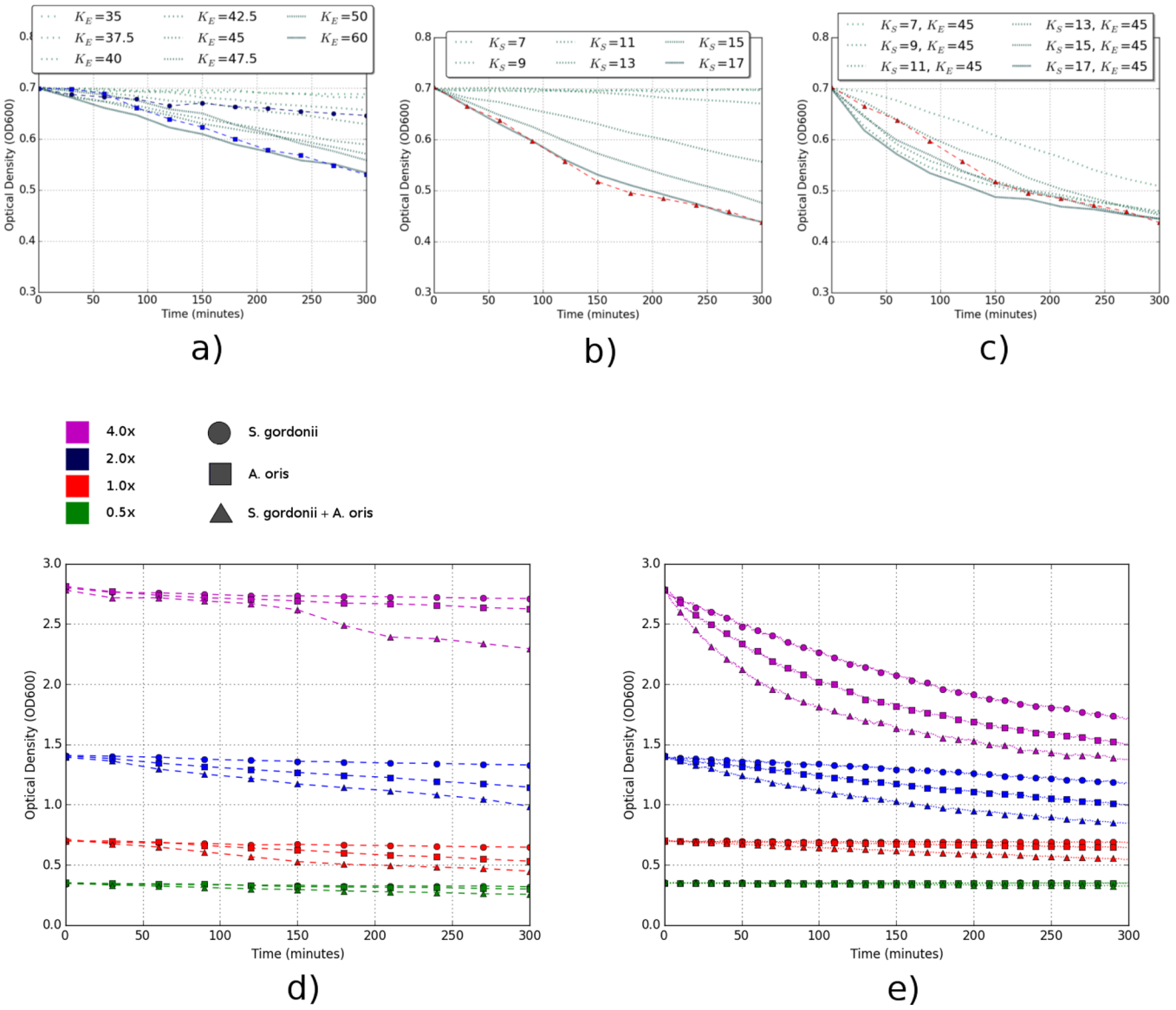


Figure 5. Simulated and experimental optical density measurements showing aggregation. Dashed lines are experimental results, solid lines are simulated. (a) Simulated aggregation due to nonspecific interactions with different force constant K_E values, compared to experimental optical density curves for single population aggregation. (b) Simulated aggregation due to specific receptor-adhesin interactions with different force constants K_S values, compared to experimental optical density curve of mixed population coaggregation. (c) Simulated aggregation due to both nonspecific and specific interactions with different force constants $K_{S/E}$ values, compared to experimental optical density curve of mixed population coaggregation. (d) Experimental optical density curves of two monoaggregating and one coaggregating population. The three curves are shown for each density (from top to bottom) 4.0×, 2.0×, 1.0× and 0.5×. (e) Simulated optical density curves of aggregation curves for *A. oris* and *S. gordonii* aggregation independently, and one curve for coaggregation of a mixed population. The three curves are shown for each density (from top to bottom) 4.0×, 2.0×, 1.0× and 0.5×.

environment at a random position adjacent to the cell. The EPS particle has the same volume as V_{EPS} and the capsule volume V_i^C is reset to 0.

EPS particles are modeled as passively motile spheres which may undergo specific and nonspecific interactions with neighboring EPS particles and cells, as described in the Physics section.

Features. *Analysis Suite.* Simbiotics has a built-in analysis suite; this consists of additional submodels that can be attached to a model specification to perform measurements. A virtual lab is implemented for more in-depth analysis, offering typical wetlab instruments and mathematical analysis features. Analysis tools and data exporters can be attached to the model specification and used to collect data and process it throughout the simulation. Users may define schedules which automate model analysis modules, programming specific model interactions or data collection and processing events.

The virtual lab currently consists of microsensors for sampling chemical field, biomass, biofilm height and gene expression profiling. A simulated spectrophotometer to obtain optical density measurements is also implemented. Virtual light is projected into a face of the cubic simulation domain, the ratio of light we detect leaving the opposite face of the domain is used to derive the optical density measurement. This is achieved by projecting the cells onto the face of the domain from which the light enters, then partitioning this into a 2D

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

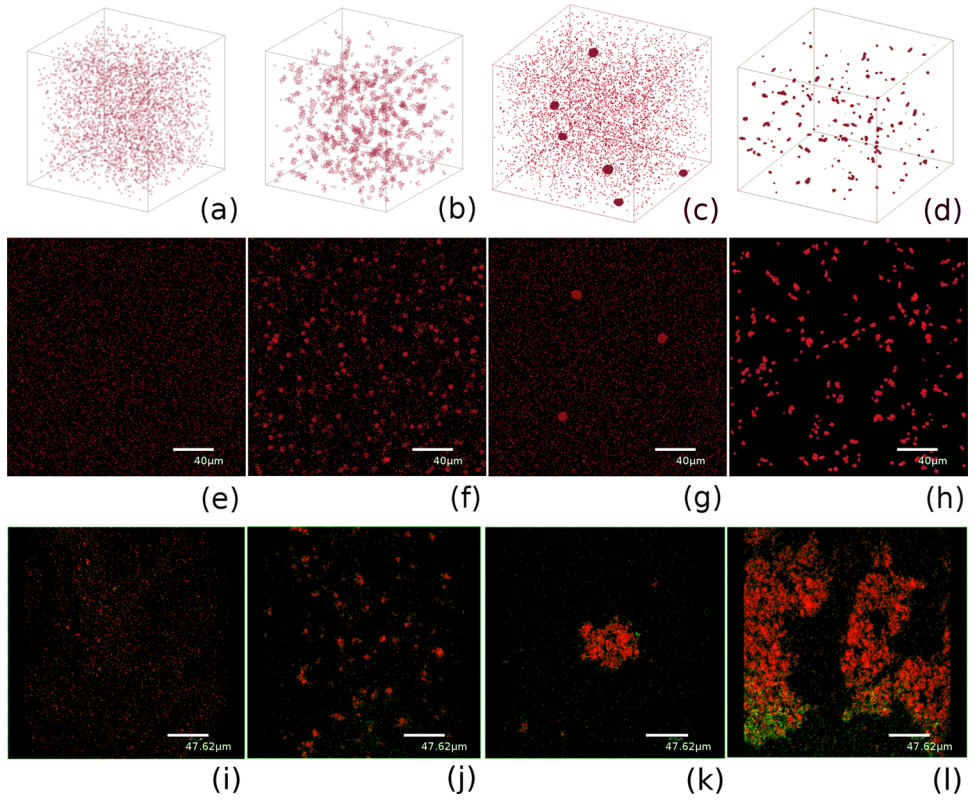


Figure 6. Simulation snapshots and microscopy images showing aggregation in both simulated and actual system. In microscopy images *A. oris* are red and *S. gordonii* are green. (a) Initially well-mixed population of simulated cells (both *A. o* and *S. g*). (b) Uniformly distributed aggregates after 4 simulated hours. (c) Large aggregates among many unbonded cells. (d) Asymmetrically distributed aggregate sizes with few unbonded cells. (e−h) 2D projections of cross sections taken from (a−d). (i) Microscopy image of experiment showing an initially well-mixed population. (j, k, l) Micropscopy images showing different aggregation structures. All images show the microscopy/simulation after 4 h.

grid. We consider the light not to pass through a grid voxel if a cell intersects it.

Additional statistics tools include measurements of the mean squared displacement and velocity autocorrelation function of bacteria, as well as detailed data gathering regarding cell interactions, gene expression and spatially distributed biomass concentrations. One can also run a biofilm height measurement algorithm that can encode a heatmap image of biofilm heights as well as measuring the average and standard deviation of measurements. A general data collector is implemented, which allows the modeler to append desired properties of the system they wish to know, such as cell species number, chemical concentrations, simulation execution time, gene expression and number of cell−cell interactions.

An optical real time 3D rendering is provided by the interface, this allows for the custom rendering of different model components. Live graph-plotting is available to show model statistics during simulations. Snapshots and videos of the simulation can be taken, with optional filters to allow for Z-stack slices, filtered cell populations or cell state highlighting. Snapshots consist of all agent geometry encodings and user-selected states, they may be loaded back into Simbiotics which reconstructs the physical state and allows for the navigation of the 3D model. Additionally a basic PovRay exporter can convert a Simbiotics snapshot into a PovRay image file to be rendered.

All lab modules may be attached to a model specification in the same way modules are attached to describe system dynamics. Modules have parameters for users to tune their behavior. Characterization of systems using the virtual lab may be achieved through parameter sweeps. The user may set a model parameter to be a sweep, such that the simulation will run multiple version each with a different parameter value in the sweep.

Microscopy Image Processing. A major contribution of Simbiotics is the ability to process microscopy images of 2D and 3D bacterial conformations. This allows for the initialization of simulations from realistic biological configurations.

To initialize the spatial configuration of bacteria one may use microscopy Z-stack images. Image processing requires three steps: we first apply a theshold image sementation that generates binary data representing the Z-stack, we then identified individual cells and encode details such as center position and radius in a data file. The data file can then be loaded in the configuration file that Simbiotics uses to initialize the cell population. This process is depicted in Figure 3 where a Z-stack image is loaded into Simbiotics.

For a multispecies population one may use image analysis techniques to identify cell species, for visually similar species one may use staining techniques to differentiate. Once the cell species has been identified this can be used to attach relevant model processes describing the cell's behavior, such as an SBML model and other Simbiotics submodels.

This microscopy image processing allows for the simulation of an imaged population, as well as the simulation of a subset of the population through some filtering process. Through this one may observe the effect the filtered subpopulation has on the development of the population by the divergence of the filtered model from the original.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

One may also compare the simulated state and the experimental state as the system evolves, iteratively changing model parameters to fit them to the experimental dynamics. This process could be automated, allowing for parameter fitting of models through refinement of the original specification based on the actual data.

CASE STUDIES

Simbiotics has been used to pursue two experimental case studies that are presented here. The first relates to oral bacterial aggregation in a fluid, and the second relates to *E. coli* biofilm formation. We develop a model of the experimental system, then explore the effect that model parameters have on the system dynamics.

Bacterial Coaggregation in a Fluid. Bacterial cells can aggregate together and form clusters. This process is governed by the surface characteristics of cells that dictate the forces they exert on each other. Cell surfaces may be charged, this affects the strength of van der Waals forces and electrostatic repulsion, known as nonspecific interactions. Surface adhesins and receptors may also be present, which undergo specific interactions if they have the appropriate structure to form an adhesive bond, this bond has a key-lock mechanism.[54] We consider the influence of these surface proteins in the aggregation of two bacteria found in the mouth, *Streptococcus gordonii* and *Actinomyces oris*, which have a matching adhesin and receptor pair (Figure 4).[55]

Single species aggregation is governed by nonspecific surface interactions as there are no matching adhesin-receptor pairs. Coaggregation of a mixed population is facilitated by a matching adhesin-receptor on the surfaces of *S. gordonii* and *A. oris*. To isolate the process of surface-mediated interactions without metabolic behavior, the cells were initially washed in sodium azide, such that their biological activity is ceased but their physical properties where preserved. The aggregation of bacteria in a cuvette of 1 mL solution was measured by following changes in optical density. We started with a well-mixed population and use a spectrophotometer to obtain a time series of OD_{600} measurements. As aggregates formed the optical density of the population decreased as more light could pass through the cuvette.

Analysis. To understand the dynamics of coaggregation we first isolate the processes of nonspecific mediated monoaggregation. We then consider the indepedent effect of a specific interaction between two species. The combined model of nonspecific and specific interactions are then analyzed. Finally we consider the effect of cell population density on the system, performing experiments and simulations of mono and coaggregation at 3 different initial densities, 0.5×, 2.0× and 4.0× that of the original system.

Analysis of nonspecific interactions involved changing the force bacteria exert on each other capturing different surface charges. Figure 5 (a) shows optical density measurements for both the experimental and simulated tests. The experimental curves show the single-species aggregation behavior of *S. gordonii* and *A. oris* on their own. As the nonspecific interaction force constant K_E is increased the rate at which aggregates form increases, however it saturates at values of $K_E > 50$. With high force constant values regularly sized aggregates typically form as seen in Figure 6 (b), low force constant values lead to irregular aggregation at around $K_E = 35$ as seen in Figure 6 (d). Aggregation does not occur when $K_E \leq 30$, this is due to the attractive electrostatic force not being sufficient to prevent Brownian motion from causing the cells to dissociate.

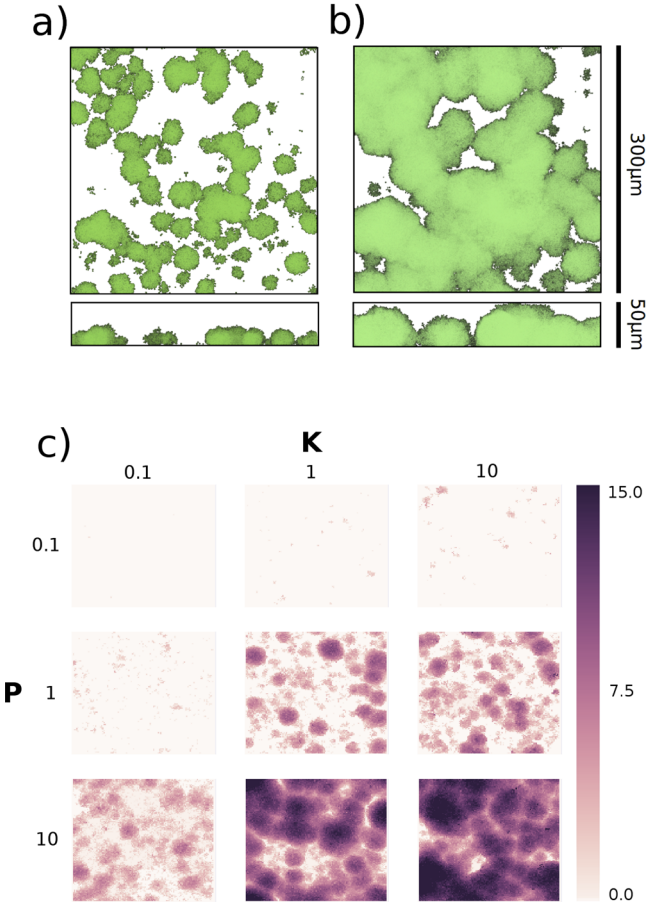


Figure 7. Simulated biofilm with $K = 10$ and $P = 10$, shown after 12 h in (a) and after 24 h in (b). Both have a cross-section displayed displayed below. (c) Parameter sweep of K and P. Nine heatmaps of simulated biofilm height measurements (μm) with different K and P parameters where cell−surface and cell−cell interactions have the same rates. These images show the biofilm after 12 h.

Analysis of additional specific interactions involved changing the probability P_S at which two colliding bacteria with matching receptor and adhesin will interact specifically, and the strength K_S of this interaction. Figure 5 (b) shows optical density measurements for experimental and simulated coaggregation. Experimental (dashed lines) show coaggregation of a mixed *S. gordonii* and *A. oris* population. Simulated (solid lines) show coaggregation optical density measurements, a parameter sweep over K_S and P_S was performed. Figure 5 (c) shows the same experimental results with the coaggregation results of the combined specific and nonspecific interaction model. One can see the aggregation rates from purely specific interactions as seen in Figure 5 (b) are enhanced by the presence of nonspecific interactions as seen in Figure 5 (c). An explanation for why this may occur is due to nonspecific interactions having an extended sphere of influence, interacting with neighbors which are not in direct contact, whereas specific interactions only occur for cells which are in physical contact.

Strong specific interactions typically lead to uniform aggregate sizes similar to nonspecific interactions, however variations of low K_S and P_S values lead to irregular aggregation as seen in Figure 6 (c) and (d). The reason for this may be due to the fact that single cell−cell interactions easily dissociate due

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

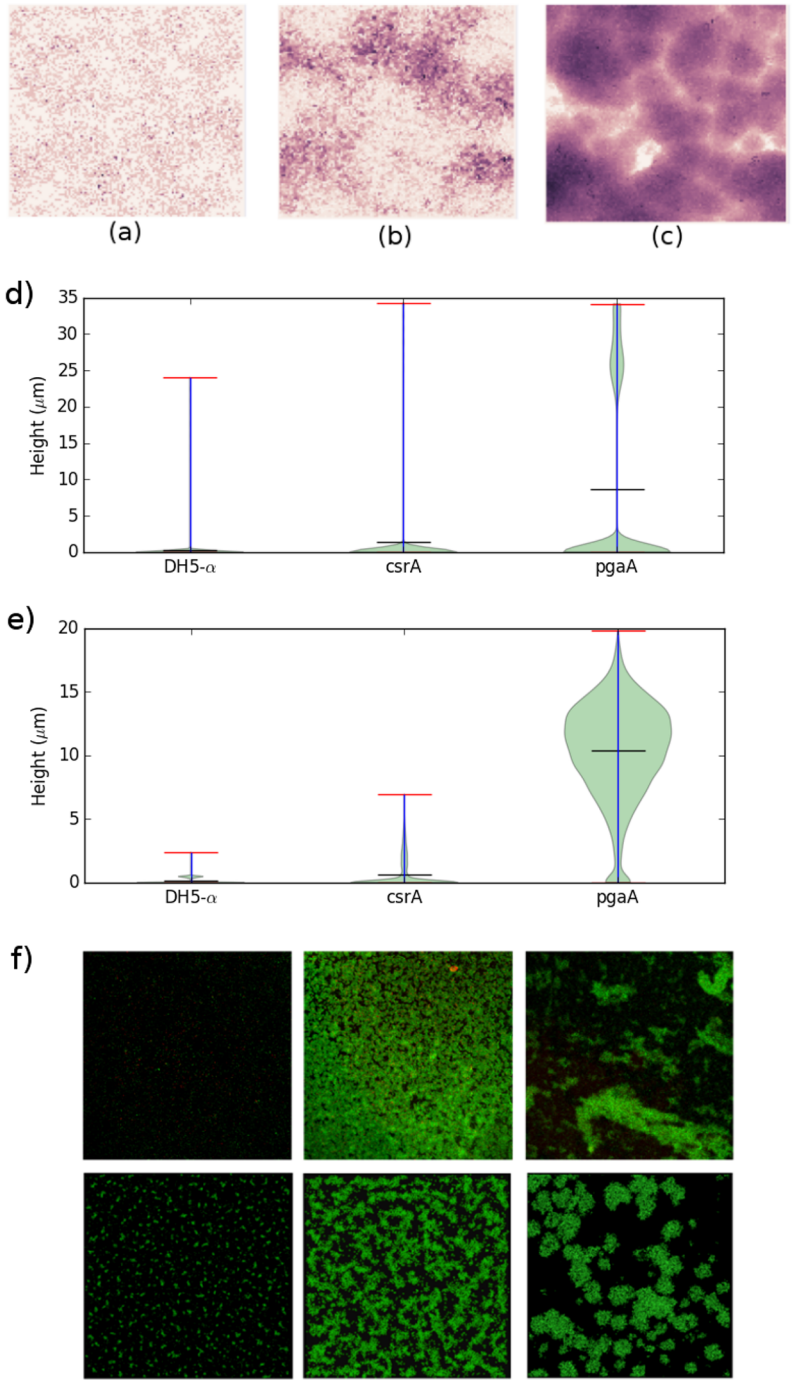


Figure 8. (a) Biofilm grown with high cell−surface interaction rates and low cell−cell interaction rates [K_S = 10, P_S = 10, K_C = 0.1, P_C = 0.1]. (b) Biofilm grown with high cell−surface interaction rates and medium cell−cell interaction rates [K_S = 10, P_S = 10, K_C = 0.25, P_C = 0.25]. (c) Biofilm as seen in Figure 7 (c) [K = 10, P = 10] which has equal K and P values for both cell−cell and cell−surface. (d) Experimental biofilm height distribution. (e) Simulated biofilm height distribution. (f) Comparison between microscopy images and simulated synthetic of biofilms formed by *E. coli* strains. Top row shows microscopy images of *DH5−α*, *csrA* and *pgaA* in order. The bottom rows shows snapshots of simulated biofilms for corresponding strains.

to Brownian motion, however if small aggregates persist, other cells may join to form a larger aggregate. The presence of additional cells creates a network of specific interactions between neighboring cells in the aggregate. A synergistic effect occurs where each cell has multiple interactions, stabilizing the aggregate and leading to a few large aggregates forming in a generally well-mixed population. When interactions are strong enough, cells do not dissociate once they adhere to each other. In a well-mixed solution with uniformly distributed Brownian motion this leads to regularly sized aggregates.

Our model does not produce the large aggregate islands as seen in the microscopy images shown in Figure 6 (h). The reason for this may be due to additional forces present in the actual experiment such as hydrodynamics which we do not model explicitly.

Aggregation at Varied Population Densities. To consider the effect of population density on aggregation experiments

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

with 0.5×, 2× and 4× the initial population size were performed. Figure 5 (d) shows experimental results, and (e) shows simulation results.

We find the aggregation rate of bacteria is proportional to the population density. This can be explained by the mean free path that a cell travels before interacting with another cell decreasing as density increases, therefore a higher probability of a physical adhesion as density increases. An additional mechanism is that large clusters tend to sink faster, as the combined motions of its constituent cells cancel, leading to a stronger effect of gravity. The sinking of aggregates leads to a decrease in the OD_{600} reading.

We find that aggregation rate is more sensitive to population density in the simulation, however overall simulation results show strong qualitative trends with the experimental findings. We note that spectrophotometry is a generalized technique which has been applied to the measurement of aggregation, and due to the nature of the method it is not a perfect deduction of aggregate formation.

Biofilm Development. Biofilms are colonies of bacteria which have adhered irreversibly to a substratum, this process is governed by complex metabolic and genetic adaptions which effect the characteristics of a cell such as surface properties and motility.[56,57] Surface colonisation is heavily influenced by surface characteristics of cells effecting their adhesion to the surface and other cells.[5,58] Further architectural development of the biofilm is thus effected by the pattern of initial attachment points to the surface, as well as the spreading pattern of the growing colony due to cell−cell interactions. We consider mutant *E. coli* forming biofilm, and aim to understand the effect of cell surface properties on biofilm development. Three strains of *E. coli* are used in the experiments, *DH5-α*, a *csrA* strain with higher surface charge, and a *PgaA* strain with an even higher surface charge.

The strains were cultivated overnight for 16 h in a 3 mL Synthetic Urine media with the addition of 0.1% glucose[59] aerobically at 30 °C and 120 rpm. Overnight grown cultures were then reinoculated into fresh Synthetic Urine media (1:100 dilution) and 200 μL was grown in a 96 well plate in the static condition for 48 h. The supernatant was then removed and its optical density was measured. The optical density of the biofilm formed on the surface was also measured by resuspending the biofilm with the synthetic urine media, and the planktonic/biofilm ratio was considered. The biofilm was also imaged by staining the biofilm formed on these 96 well plates using the Live/Dead BacLight stain (ThermoFisher Scientific, UK) using a Leica SP2 confocal laser scanning microscope.

Analysis. We consider the height distribution of biofilms to characterize their morphologies. Biofilms which are flat and uniform produce a low standard deviation in height, whereas lumpy and irregular biofilms produce a larger height variation. Through this process we can relate local cell surface interactions to colony level spatial organization.

We observe the effect of cell surface charge by growing biofilms with different cell parameters. The parameters modified are the rate P_S with which a cell adheres to a surface it is in contact with, strength K_S of the interaction with the surface, rate P_C with which a cell will adhere to another contacting cell and strength K_C of that cell−cell interaction. Snapshots of a simulated biofilm can be seen in Figure (a) and (c).

First we set cell−surface and cell−cell parameters to be symmetric, such that cells have the same rate at which they interact with other cells and surfaces, and they interact with other cells and surfaces with the same force constant.

Table 3. Parameters for the Coaggregation Case Study Model

submodel	variable	symbol	value	unit
Sphere	*S. gordonii* cell radius	$r_{gordonii}$	0.5	μm
Sphere	*A. oris* cell radius	r_{oris}	0.5	μm
Brownian motion	Force constant	K_R	2.2	$\frac{\mu m}{cs^{3/2}}$
Friction	Force constant	K_F	2.0	$\frac{\mu g}{cs}$
Gravity	Force constant	K_G	0.0002	$\frac{\mu m}{cs^2}$
Nonspecific interactions	Force constant	K_E	25−50	$\mu g \frac{\mu m^3}{cs^2}$
	Range	r_E	3.0	$\frac{\text{interactions}}{cs}$
Specific interactions	Force constant	K_S	6−7	$\frac{\mu g}{cs^2}$
	Probability	P_S	0.1−10	$\frac{\text{interactions}}{cs}$

From Figure 7 (c) one can see that as we increase the rate at which cells interact $P_{S/C}$, the biofilm covers more surface area due to more bacteria attaching directed to the substratum. Clusters then form as other planktonic cells attach to those already in the biofilm. As we increase the strength of cell interactions $K_{S/C}$ we observe taller and denser biofilms, this may be explained by the fact that bacteria stick to each other more firmly and thus the biofilm can grow stable mushroom-like structures which extend from the substratum into the fluid medium.

By changing the parameters $P_{S/C}$ and $K_{S/C}$ we obtain varied biofilm development. However, these parameters produce a consistent biofilm architecture, with hemispherical clusters of bacteria spreading across the surface forming lumpy and irregular biofilms.

We consider the effect that an asymmetrical cell adhesion to other cells than to surfaces would have. In Figure 8 (a, b, c) we compare biofilms grown with symmetric cell−cell and cell−surface adhesion to a biofilm grown with asymmetric parameters, such that the probability a cell will bind to a surface P_S and the strength of that cell−surface interaction K_S are relatively high in comparison to cell−cell interaction probability P_C and strength K_C. This results in significantly reduced biofilm formation, with a spreading of cells across the surface leading to a more uniform structure.

Our model and analysis offer an explanation as to how cell−surface interactions can influence biofilm architecture. When cells interact with the environmental surfaces and other cells at a similar rate biofilms tend to form an irregular and lumpy structure. This can be explained by early colonisation leading to clustered growth on the surface forming an irregular structure, as lumps increase the probability that planktonic cells will adhere to them as they protrude into the fluid due to strong cell−cell interactions. Cells that interact weakly with each other but strongly with a surface tend to form flat and uniform biofilms. This can be explained by cells in the biofilm being able to detach from other cells, allowing them to spread across the surface or becoming planktonic in the fluid; they may then colonise the surface elsewhere. Over time cells populate the surface, but due to weak cell−cell interactions a thick layer of cells does not emerge until the surface is covered forcing growth in height.

Model findings reinforce the observations made in experiments, Figure 8 (d) shows experimental biofilm height distributions and (e) simulated height distributions. Strains such as *PgaA* which have a higher surface charge have stronger cell−cell interactions

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

which lead to large irregular biofilms, where as a low surface charge strain *DH5-α* produce less biofilm with a uniform structure. Visual comparisons between experimental and modeled biofilms can be seen in Figure 8 (f).

Table 4. Parameters for the Biofilm Case Study Model

submodel	variable	symbol	value	unit
Sphere	*E. coli* cell diameter	r	1.0	μm
Brownian motion	Force constant	K_{Rp}	2.2	$\frac{\mu m^{3/2}}{cs}$
	Force constant	K_{Rs}	1.0	$\frac{\mu m}{cs^{3/2}}$
Friction	Force constant	K_F	2.0	$\frac{\mu g}{cs}$
Gravity	Force constant	K_G	0.0002	$\frac{\mu m}{cs^2}$
Cell growth	Growth rate	G_R	0.00025	$\frac{\mu g}{cs}$
Specific interactions	Cell−cell force constant	K_C	0.1−10	$\frac{\mu g}{cs^2}$
	Cell−surface force constant	K_S	0.1−10	$\frac{\mu g}{cs^2}$
	Cell−cell probability	P_C	0.1−10	$\frac{\text{interactions}}{cs}$
	Cell−surface probability	P_S	0.1−10	$\frac{\text{interactions}}{cs}$

DISCUSSION

In this article, we have presented Simbiotics, a novel individual based simulator for bacterial populations and biofilms, and demonstrated its modeling and analysis features alongside two example case studies that illustrate its potential for computer aided design in the synthetic biology workflow. The Simbiotics library provides a range of submodels that can be attached to a model specification to represent the target system, as demonstrated in the case studies.

There are numerous simulators for population dynamics, however there is yet to be a standardized platform for modeling bacterial populations in a multiscale manner. Simbiotics provides an extendable modular framework in which the user can integrate a wide range of processes, including interfacing with standard formats such as SBML for modeling individual cells and microscopy images for describing spatial composition of populations. The extendable library and distributed CPU parallelization features allow for the scaling of Simbiotics functionality as it is further developed.

Our case studies focus on physical properties of the modeled systems. Although Simbiotics also allows for the detailed

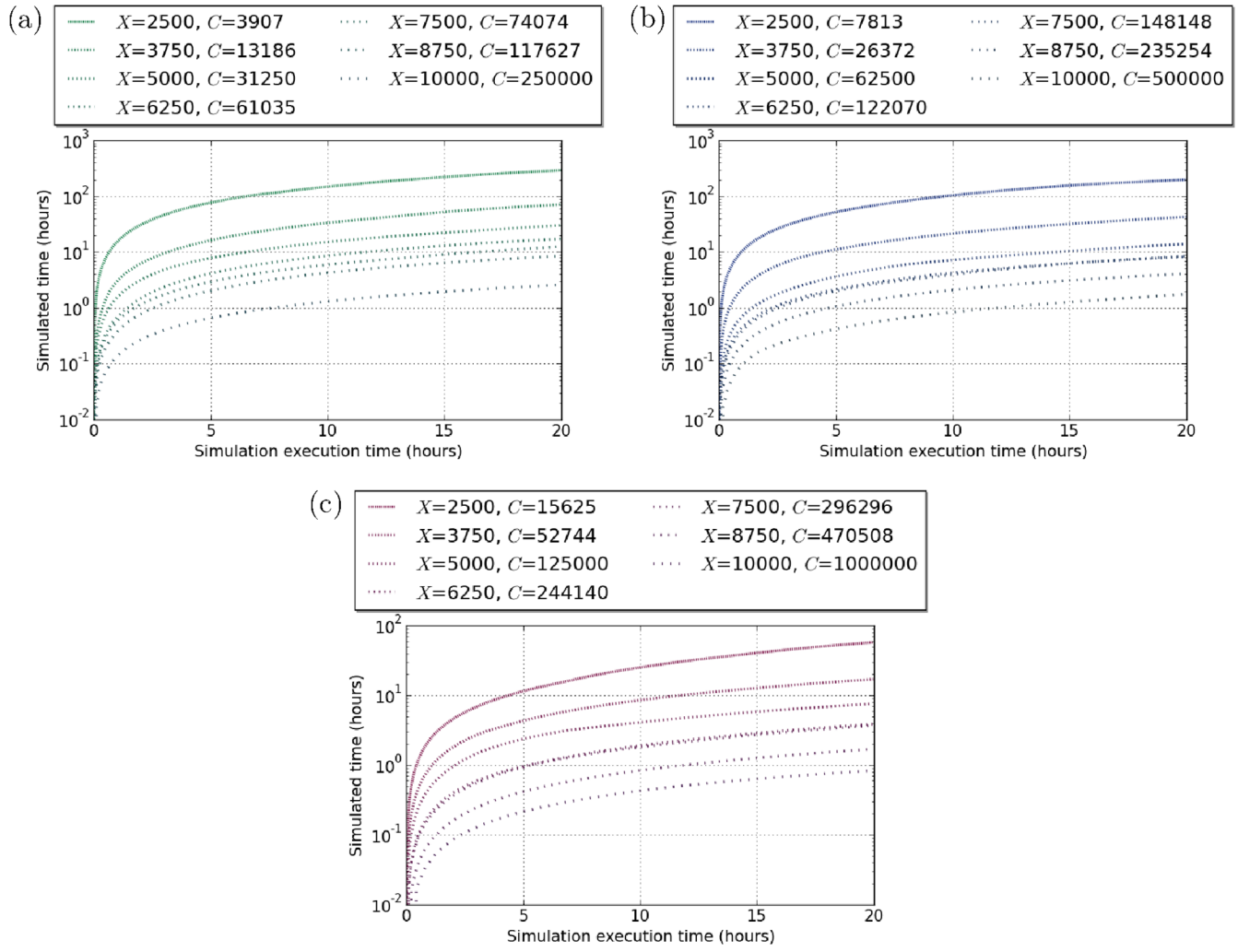


Figure 9. Stress test showing log of simulated time against the simulation execution time. Stress tests were performed with spherical cells, suspended in a fluid experiencing a random mixing force. For each test, C is the number of cells, and X is the length of one side of the cubic simulation domain in μm. (a) Results for a density of $\frac{1.25 \times 10^5 \text{ cells}}{\text{mL}}$. (b) Results for a density of $\frac{5 \times 10^5 \text{ cells}}{\text{mL}}$. (c) Results for a density of $\frac{1 \times 10^6 \text{ cells}}{\text{mL}}$. All tests were performed on two cores of a single node of the HPC.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

modeling and characterization of cell internal/biological processes, we have used this ability only to a limited degree in these studies. This choice was guided by the observation that there are already numerous tools for the biodesign automation of genetic circuits[38,60] and metabolic pathways[61,62] but that there is still a lack of physically accurate population level simulators which can integrate these submodels.

Case study findings show Simbiotics' flexibility in modeling a target system, and the process with which model specifications can be refined to further understand the dynamics of a target system. Model development can verify that our understanding of the experimental system is correct, and explain the driving forces behind population behavior. It can also expose discrepancies between the real system and the simulated one, thereby revealing areas of insufficient system understanding.

We find that interactions between cells lead to statistical population behavior that is comparable to experimental results. Spatial organization is not only guided by physical interactions between cells, but also by biochemical properties which may modulate the way in which cells interact.

In future work, we plan to expand the Simbiotics modeling library and virtual lab, develop a graphical user interface, alongside optimization of the simulation core. Furthermore, we also plan to integrate Simbiotics with our in-house Infobiotics 2.0 design suite for synthetic genetic designs,[20] as well as the SBOL data format, which provides a standard for representing and communicating synthetic biology systems.[63]

Data Plotting. All data plots were generated using custom scripts using Python 2.7.9, Matplotlib 1.4.2 and Seaborn 0.6.0. Data was automatically exported from Simbiotics and into Python scripts.

Simbiotics Coaggregation Model. To model the dynamics of the system we approximate bacterial cells to be spheres with surface properties. Each cell performs a random-walk due to the effect of Brownian motion causing the population to mix. Cells have an extended sphere of influence to represent their surface charge effects, these are modeled as nonspecific equations as described in the modeling section. *S. gordonii* cells have adhesins on their surface and *A. oris* have a matching receptor, an interaction between the two is modeled as described in the modeling section under specific interactions.

As the main parameters to the simulation we consider the strength of nonspecific interactions due to surface charge, K_E. We also take the probability that two colliding cells with a matching receptor-adhesin will interact P_S, representing the density of adhesins and receptors on the cell surfaces. Further more we consider the strength of an adhesin-receptor interaction K_S. All of the model parameters are displayed in Table 3.

We start with a well mixed population of individual bacteria and use the simulated spectrophotometer as described in the Simbiotics Analysis section to obtain a timeseries of optical density measurements.

Simbiotics Biofilm Model. Our model of biofilm development represents cell growth, cell motility, surface-mediated interactions and basic gene regulation. Model initialization starts with a planktonic population suspended in the fluid phase, with a solid substratum existing on the bottom face of the simulation domain. Cells experience Brownian motion with a force constant K_{Rp}, and may come into contact with the substratum. Cells contacting the substratum may adhere to it with the rate P_S and an interation strength K_S. Once attached to the substratum, cells experience a lower magnitude of Brownian motion, K_{Rs}, and may adhere to other contacting cells at a rate P_C and an interaction strength K_C. These cells are then considered to be part of the biofilm and experience lower Brownian motion and may adhere to other cells. Cells grow at a rate G_R with growth dynamics as described in the Simbiotics modeling section. All model parameters can be seen in Table 4.

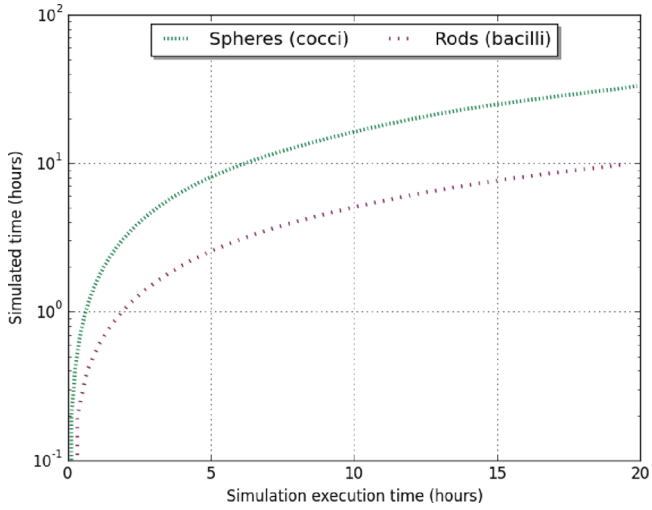


Figure 10. A stress test of sphere and rod-shaped cells, showing the log of simulated time against the simulation execution time (both in hours). 100 000 spherical and 100 000 rod-shaped cells were simulated independently, suspended in a fluid volume of 1 mL. Cells experience a random mixing force to induce collisions. Tests were performed on two cores of a single node of the HPC.

COMPUTATIONAL DETAILS

The bacterial coaggregation case study model involved simulating between 20 000 and 160 000 cellular agents in a cubic simulation domain of length 368 μm. Simulating 5 h of aggregation took between 4 and 38 h of computational time.

The biofilm case study model involved simulating an initial population of 3000 cellular agents in a cuboidal domain of size 300 × 50 × 300 μm. Simulating 12 h of biofilm growth took 20 h of computational time, at this time there were 750 000 cellular agents in the simulation domain.

Both case study models were simulated on a high performance computing cluster. The simulations were run on one node, multithreaded across 5 cores. The node CPU specification being Xeon E5−2690 v2 with 256GB of RAM.

Performance analysis shows that calculating the physics of the system such as cell−cell collisions, movement and interaction forces, consume the largest amount of computational time. The scaling of the physics calculations is linear with the number of cells $O(N)$, however becomes $O(N^2)$ when cells are packed together extremely closely such as when aggregates form or dense biofilms.

Stress Tests. We preformed stress tests on the Simbiotics platform, analyzing the performance scaling for different population and domain sizes. Tests were performed on two cores of a single node of the HPC (Figure 9).

Performance Scaling. First we tested how performance scales for cellular populations of the same density for varied domain volumes. We performed the tests for 3 population densities, $\frac{1.25\times10^5\ \text{cells}}{\text{mL}}$, $\frac{5\times10^5\ \text{cells}}{\text{mL}}$ and $\frac{1\times10^6\ \text{cells}}{\text{mL}}$. For each density we scale the population and domain size, maintaing the same density and observing how performance scaled.

Sphere and Rod Comparison. See Figure 10 for details.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acssynbio.6b00315.

Computational details (PDF)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: natalio.krasnogor@newcastle.ac.uk.

ORCID

Harold Fellermann: 0000-0001-5861-1945

Phillip C. Wright: 0000-0002-8834-0426

Natalio Krasnogor: 0000-0002-2651-4320

Author Contributions

JN wrote the manuscript, developed the software and models and performed data analysis. HF supervised the work, designed the overall study, specified the software functionality, and curated the manuscript. YD developed the microscopy image processing module. WM performed experiments for coaggregation study. NJ supervised coaggregation study and curated the manuscript. JM performed experiments for biofilm study and wrote the experimental description. CB supervised the biofilm study. PW curated the manuscript. NK directed and supervised the work, designed the overall study, specified the software functionality, edited the manuscript and secured funding.

Notes

The authors declare no competing financial interest.

The Simbiotics software and accompanying User Manual are available at http://ico2s.org/software/simbiotics.html The manual explains how to download, install and use Simbiotics, alongside model building tutorials. Additionally there is a small gallery of models and images.

■ ACKNOWLEDGMENTS

This work was supported by EPSRC grants EP/I031642/2, EP/J004111/2, EP/L00149012/2, EP/N031962/1 and the MRC (MR/N005872/1). We also acknowledge Newcastle University and the School of Computing Science for granting access to the high performance computing cluster on which the simulations were executed.

■ REFERENCES

(1) Grosberg, R. K., and Strathmann, R. R. (2007) The Evolution of Multicellularity: A Minor Major Transition? *Annual Review of Ecology, Evolution, and Systematics 38*, 621−654.

(2) Federle, M. J., and Bassler, B. L. (2003) Interspecies communication in bacteria. *J. Clin. Invest. 112*, 1291−1299.

(3) Waters, C. M., and Bassler, B. L. (2005) QUORUM SENSING: Cell-to-Cell Communication in Bacteria. *Annu. Rev. Cell Dev. Biol. 21*, 319−346.

(4) Kreft, J.-U. (1999) Conflicts of interest in biofilms. *Biofilms 1*, 265−276.

(5) O'Toole, G. A., and Wong, G. C. (2016) Sensational biofilms: surface sensing in bacteria. *Curr. Opin. Microbiol. 30*, 139−146.

(6) Kolari, M., Nuutinen, J., and Salkinoja-Salonen, M. S. (2001) Mechanisms of biofilm formation in paper machine by Bacillus species: the role of Deinococcus geothermalis. *J. Ind. Microbiol. Biotechnol. 27*, 343−351.

(7) Xavier, R. S., Omar, N., and de Castro, L. N. (2011) Bacterial colony: Information processing and computational behavior. *2011 Third World Congress on Nature and Biologically Inspired Computing*.

(8) Toyoshima, M., Aikawa, S., Yamagishi, T., Kondo, A., and Kawai, H. (2015) A pilot-scale floating closed culture system for the multicellular cyanobacterium Arthrospira platensis NIES-39. *J. Appl. Phycol. 27*, 2191−2202.

(9) Marsh, P. D. (2006) Dental plaque as a biofilm and a microbial community - implications for health and disease. *BMC Oral Health 6*, S14.

(10) Pascalie, J., Potier, M., Kowaliw, T., Giavitto, J.-L., Michel, O., Spicher, A., and Doursat, R. (2016) Developmental Design of Synthetic Bacterial Architectures by Morphogenetic Engineering. *ACS Synth. Biol. 5*, 842−861.

(11) Jiang, W., Bikard, D., Cox, D., Zhang, F., and Marraffini, L. A. (2013) RNA-guided editing of bacterial genomes using CRISPR-Cas systems. *Nat. Biotechnol. 31*, 233−239.

(12) Amos, M., Axmann, I. M., Blüthgen, N., de la Cruz, F., Jaramillo, A., Rodriguez-Paton, A., and Simmel, F. (2015) Bacterial computing with engineered populations. *Philos. Trans. R. Soc., A 373*, 20140218.

(13) Purnick, P. E. M., and Weiss, R. (2009) The second wave of synthetic biology: from modules to systems. *Nat. Rev. Mol. Cell Biol. 10*, 410−422.

(14) Song, H.-S., and Liu, C. (2015) Dynamic Metabolic Modeling of Denitrifying Bacterial Growth: The Cybernetic Approach. *Ind. Eng. Chem. Res. 54*, 10221−10227.

(15) Donlan, R. M. (2002) Biofilms: Microbial Life on Surfaces. *Emerging Infect. Dis. 8*, 881−890.

(16) Garrett, T. R., Bhakoo, M., and Zhang, Z. (2008) Bacterial adhesion and biofilms on surfaces. *Prog. Nat. Sci. 18*, 1049−1056.

(17) Elowitz, M. B. (2002) Stochastic Gene Expression in a Single Cell. *Science 297*, 1183−1186.

(18) Konur, S., Fellermann, H., Marian Mierla, L., Sanassy, D., Ladroue, C., Kalvala, S., Gheorghe, M., and Krasnogor, N. (2017) In *Advances in Unconventional Computing: Vol. 2 Prototypes, Models and Algorithms* (Adamatzky, A., Ed.), pp 655−676, Springer International Publishing.

(19) Sanassy, D., Fellermann, H., Krasnogor, N., Konur, S., Mierla, L. M., Gheorghe, M., Ladroue, C., and Kalvala, S. (2014) Modelling and Stochastic Simulation of Synthetic Biological Boolean Gates. *2014 IEEE Intl Conf on High Performance Computing and Communications, 2014 IEEE 6th Intl Symp on Cyberspace Safety and Security, 2014 IEEE 11th Intl Conf on Embedded Software and Syst (HPCC, CSS, ICESS)*.

(20) Blakes, J., Twycross, J., Romero-Campero, F. J., and Krasnogor, N. (2011) The Infobiotics Workbench: an integrated in silico modelling platform for Systems and Synthetic Biology. *Bioinformatics 27*, 3323−3324.

(21) Jang, S. S., Oishi, K. T., Egbert, R. G., and Klavins, E. (2012) Specification and Simulation of Synthetic Multicelled Behaviors. *ACS Synth. Biol. 1*, 365−374.

(22) Rudge, T. J., Steiner, P. J., Phillips, A., and Haseloff, J. (2012) Computational Modeling of Synthetic Microbial Biofilms. *ACS Synth. Biol. 1*, 345−352.

(23) Gorochowski, T. E., Matyjaszkiewicz, A., Todd, T., Oak, N., Kowalska, K., Reid, S., Tsaneva-Atanasova, K. T., Savery, N. J., Grierson, C. S., and di Bernardo, M. (2012) BSim: An Agent-Based Tool for Modeling Bacterial Populations in Systems and Synthetic Biology. *PLoS One 7*, e42790.

(24) Lardon, L. A., Merkey, B. V., Martins, S., Dötsch, A., Picioreanu, C., Kreft, J.-U., and Smets, B. F. (2011) iDynoMiCS: next-generation individual-based modelling of biofilms. *Environ. Microbiol. 13*, 2416−2434.

(25) Kreft, J.-U., Booth, G., and Wimpenny, J. W. T. (1998) BacSim, a simulator for individual-based modelling of bacterial colony growth. *Microbiology 144*, 3275−3287.

(26) Goñi-Moreno, A., and Amos, M. (2015) DiSCUS: A Simulation Platform for Conjugation Computing. In *Unconventional Computation and Natural Computation: 14th International Conference, UCNC 2015, Auckland, New Zealand, August 30 − September 3, 2015, Proceedings* (Calude, C. S., and Dinneen, M. J., Eds.), Chapter DiSCUS: A Simulation Platform for Conjugation Computing, pp 181−191, Springer International Publishing.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX

(27) Zubler, F. (2009) A framework for modeling the growth and development of neurons and networks. *Front. Comput. Neurosci.*, DOI: 10.3389/neuro.10.025.2009.
(28) Newman, T. J., and Grima, R. (2004) Many-body theory of chemotactic cell-cell interactions. *Physical Review E*, DOI: 10.1103/PhysRevE.70.051916.
(29) Janulevicius, A., van Loosdrecht, M. C., Simone, A., and Picioreanu, C. (2010) Cell Flexibility Affects the Alignment of Model Myxobacteria. *Biophys. J. 99*, 3129−3138.
(30) Ghosh, P., Mondal, J., Ben-Jacob, E., and Levine, H. (2015) Mechanically-driven phase separation in a growing bacterial colony. *Proc. Natl. Acad. Sci. U. S. A. 112*, E2166−E2173.
(31) Farrell, F. D. C., Hallatschek, O., Marenduzzo, D., and Waclaw, B. (2013) Mechanically Driven Growth of Quasi-Two-Dimensional Microbial Colonies. *Phys. Rev. Lett.*, DOI: 10.1103/PhysRevLett.111.168101.
(32) Lifshitz, E., Kosevich, A., and Pitaevskii, L., Eds. (1986) *Theory of Elasticity*, 3rd ed., p ii, Butterworth-Heinemann, Oxford.
(33) Hermansson, M. (1999) The DLVO theory in microbial adhesion. *Colloids Surf., B 14*, 105−119.
(34) Ninham, B. (1999) On progress in forces since the DLVO theory. *Adv. Colloid Interface Sci. 83*, 1−17.
(35) Barth, T., and Ohlberger, M. (2004) Finite Volume Methods: Foundation and Analysis. In *Encyclopedia of Computational Mechanics*.
(36) de Jong, H. (2002) Modeling and Simulation of Genetic Regulatory Systems: A Literature Review. *J. Comput. Biol. 9*, 67−103.
(37) Conrad, E. D., and Tyson, J. J. (2006) In *System Modeling in Cellular Biology: From Concepts to Nuts and Bolts* (Szallasi, Z., Stelling, J., and Periwal, V., Eds.), Chapter 6, pp 98−123, The MIT Press.
(38) Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., the rest of the SBML Forum, Arkin, A. P., Bornstein, B. J., Bray, D., et al. (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics 19*, 524−531.
(39) Takizawa, H., Nakamura, K., Tabira, A., Chikahara, Y., Matsui, T., Hiroi, N., and Funahashi, A. (2013) LibSBMLSim: a reference implementation of fully functional SBML simulator. *Bioinformatics 29*, 1474−1476.
(40) Cooper, S., Ed. (1991) *Bacterial Growth and Division*, p ii, Academic Press, San Diego.
(41) Chien, A.-C., Hill, N., and Levin, P. (2012) Cell Size Control in Bacteria. *Curr. Biol. 22*, R340−R349.
(42) Rothfield, L. I., and Zhao, C.-R. (1996) How Do Bacteria Decide Where to Divide? *Cell 84*, 183−186.
(43) Mir, M., Wang, Z., Shen, Z., Bednarz, M., Bashir, R., Golding, I., Prasanth, S. G., and Popescu, G. (2011) Optical measurement of cycle-dependent cell growth. *Proc. Natl. Acad. Sci. U. S. A. 108*, 13124−13129.
(44) Donachie, W. D., Begg, K. J., and Vicente, M. (1976) Cell length, cell growth and cell division. *Nature 264*, 328−333.
(45) Macey, R. I. (1986) Mathematical Models of Membrane Transport Processes. *Physiology of Membrane Disorders*, 111−131.
(46) Uzman, A. (2003) *Molecular Biology of the Cell*, 4th ed. (Alberts, B., Johnson, Lewis, J., Raff, M., Roberts, K., and Walter, P., Eds.) Biochemistry and Molecular Biology Education, Vol. *31*, pp 212−214.
(47) Tindall, M. J., Maini, P. K., Porter, S. L., and Armitage, J. P. (2008) Overview of Mathematical Approaches Used to Model Bacterial Chemotaxis II: Bacterial Populations. *Bull. Math. Biol. 70*, 1570−1607.
(48) Patteson, A. E., Gopinath, A., Goulian, M., and Arratia, P. E. (2015) Running and tumbling with E. coli in polymeric solutions. *Sci. Rep. 5*, 15761.
(49) Paoluzzi, M., Leonardo, R. D., and Angelani, L. (2013) Effective run-and-tumble dynamics of bacteria baths. *J. Phys.: Condens. Matter 25*, 415102.
(50) Keller, E. F., and Segel, L. A. (1971) Model for chemotaxis. *J. Theor. Biol. 30*, 225−234.
(51) Wang, C. C., Ng, K.-L., Chen, Y.-C., Sheu, P. C., and Tsai, J. J. (2011) Simulation of Bacterial Chemotaxis by the Random Run and Tumble Model. *2011 IEEE 11th International Conference on Bioinformatics and Bioengineering*.
(52) Sheng, G.-P., Yu, H.-Q., and Li, X.-Y. (2010) Extracellular polymeric substances (EPS) of microbial aggregates in biological wastewater treatment systems: A review. *Biotechnol. Adv. 28*, 882−894.
(53) kumar singha, T. (2012) Microbial Extracellular Polymeric Substances: Production, Isolation and Applications. *IOSR J. Pharm. 2*, 276−281.
(54) Sauer, M. M., Jakob, R. P., Eras, J., Baday, S., Eris, D., Navarra, G., Bernèche, S., Ernst, B., Maier, T., and Glockshuber, R. (2016) Catch-bond mechanism of the bacterial adhesin FimH. *Nat. Commun. 7*, 10738.
(55) Back, C., Douglas, S., Emerson, J., Nobbs, A., and Jenkinson, H. (2015) Streptococcus gordoniiDL1 adhesin SspB V-region mediates coaggregation *via* receptor polysaccharide ofActinomyces orisT14V. *Mol. Oral Microbiol. 30*, 411−424.
(56) Nguyen, V., Karunakaran, E., Collins, G., and Biggs, C. A. (2016) Physicochemical analysis of initial adhesion and biofilm formation of Methanosarcina barkeri on polymer support material. *Colloids Surf., B 143*, 518−525.
(57) Cutler, N. A., Chaput, D. L., Oliver, A. E., and Viles, H. A. (2015) The spatial organization and microbial community structure of an epilithic biofilm. *FEMS Microbiology Ecology 91*, fiu027−fiu027.
(58) Dufrêne, Y. F. (2015) Sticky microbes: forces in microbial cell adhesion. *Trends Microbiol. 23*, 376−382.
(59) Brooks, H., White, D., Wagstaff, A., and Michell, A. (1997) Evaluation of a glutamine-containing oral rehydrationsolution for the treatment of calf diarrhoea using an Escherichia coli model. *Vet. J. 153*, 163−169.
(60) Karlebach, G., and Shamir, R. (2008) Modelling and analysis of gene regulatory networks. *Nat. Rev. Mol. Cell Biol. 9*, 770−780.
(61) Durot, M., Bourguignon, P.-Y., and Schachter, V. (2009) Genome-scale models of bacterial metabolism: reconstruction and applications. *FEMS Microbiol Rev. 33*, 164−190.
(62) Karr, J. R., Sanghvi, J. C., Macklin, D. N., Arora, A., and Covert, M. W. (2013) WholeCellKB: model organism databases for comprehensive whole-cell models. *Nucleic Acids Res. 41*, D787−D792.
(63) Galdzicki, M., Clancy, K. P., Oberortner, E., Pocock, M., Quinn, J. Y., Rodriguez, C. A., Roehner, N., Wilson, M. L., Adam, L., Anderson, J. C., et al. (2014) The Synthetic Biology Open Language (SBOL) provides a community standard for communicating designs in synthetic biology. *Nat. Biotechnol. 32*, 545−550.

DOI: 10.1021/acssynbio.6b00315
ACS Synth. Biol. XXXX, XXX, XXX−XXX